

# The Project Information Management System (and more)

*A family of R&D web applications originated with a friendly bet*



Sometimes software products, even whole lines of software products, are created out of unassuming, unplanned chains of events. Such was the case with the Project Information Management System (PIMS). Technology Today talked with Robert L. Thorpe, a manager in SwRI's Space Science and Engineering Division, about the origins and evolution of a successful line of R&D web applications.

## **TT: What is PIMS, and what brought it about?**

RT: PIMS stands for the Project Information Management System. It began as a means to help engineers on NASA projects keep track of action items. It grew into a production-quality web application suite with 10 project management modules, many on-site users and two corporate deployments. It wasn't planned that way — it actually started with a bet at a staff meeting.

## **TT: Who made the bet?**

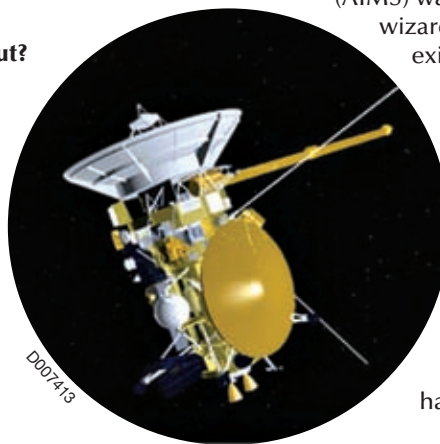
RT: I did. I was at an operations team meeting in 1998 for the Ion and Neutral Mass Spectrometer (INMS) and Cassini Plasma Spectrometer (CAPS) instruments, supporting NASA's Cassini mission to Saturn. The project manager was handing out action items to the team members, who were writing notes to themselves. It occurred to me that it would be easier for everyone if action items were tracked through a "web-application" — a new word in those days — a central web-database system, accessible to anyone with a web browser.

I proposed the idea to the project manager, who asked how long it would take. Based on some just-acquired knowledge of how to fuse software with a newly available

web server, I estimated one week.

This was not the expected answer. Based on older web technologies, weeks or months were expected, and the whole effort hung in the balance. I firmly believed the effort required just one week and repeatedly said so. In the end I was told that I had *one week*.

One week later the Action Item Management System (AIMS) was born. This was no great feat of software wizardry, but rather the combination of new and existing software technologies.



*NASA's Cassini mission to Saturn inspired the creation of the Action Item Management System.*

## **TT: Was AIMS a success?**

RT: Not right away. In fact, AIMS was used very little at first. Its biggest achievement was catching the attention of the project manager for the instrument suite on another NASA project, the New Horizons mission to Pluto.

New Horizons was proposed as a reasonably priced spacecraft that could reach the farthest planet in the Solar System. The mission was considered somewhat daring, and it had a lot of action items.

The instrument suite project manager, Bill Gibson, who is an assistant vice president in the Space Science and Engineering Division here at SwRI, needed an easy, low-cost way to track action items from formal design reviews. Each one needed to be fully addressed, completed and concurred — careful tracking and review was very important.

This was just what AIMS was designed to do. At Bill's request we deployed a New Horizons-specific AIMS server, which was christened NAIMS.

## **TT: How did NAIMS perform?**

RT: It worked very well. Project team members could access it from anywhere in the country. It provided a single,

central, web-accessible location with up-to-date information on every action item — exactly what the team needed.

An interesting nickname for NAIMS resulted from an email reminder function. NAIMS sent an email once a week to each team member that had an overdue action item. Not everyone appreciated these reminders, and one team member suggested that the system should be called “MAIMS,” for “Merciless Action Item Management System.”

Merciless or not, in the end, the action items got done.

#### **TT: What happened next?**

RT: AIMS filled a need for project managers, and soon we had five separate project-specific AIMS systems. Meanwhile, the INMS Operations Team Leader had an idea for a web-based operations planning tool to support the Cassini mission. We put forth a prototype using the same architecture and were awarded a larger contract for a full mission-level science planning system, the Cassini Information Management System (CIMS).

CIMS was a large undertaking. It required a more scalable and robust architecture than AIMS. A consultant recommended a web-application approach using Java®, HTML and Oracle®, running on any type of operating system (Unix® was preferred).

We used this architecture on CIMS, and then found ourselves using it on science data systems for the INMS and CAPS instruments on Cassini. Additionally, Bill asked us to produce Risk Management and Non-Conformance tracking web applications for project teams in the division.

#### **TT: How did you accommodate that kind of growth?**

RT: The growth was a major challenge. There were many separate systems to maintain, some using the old architecture and some under the new. More requests were coming in.

We received some seed money from SwRI and created a single multi-project AIMS system, using the new architecture. This system had security controls to prevent users from accessing data on other projects. Then we folded in the Risk and Non-Conformance systems, and PIMS was born.

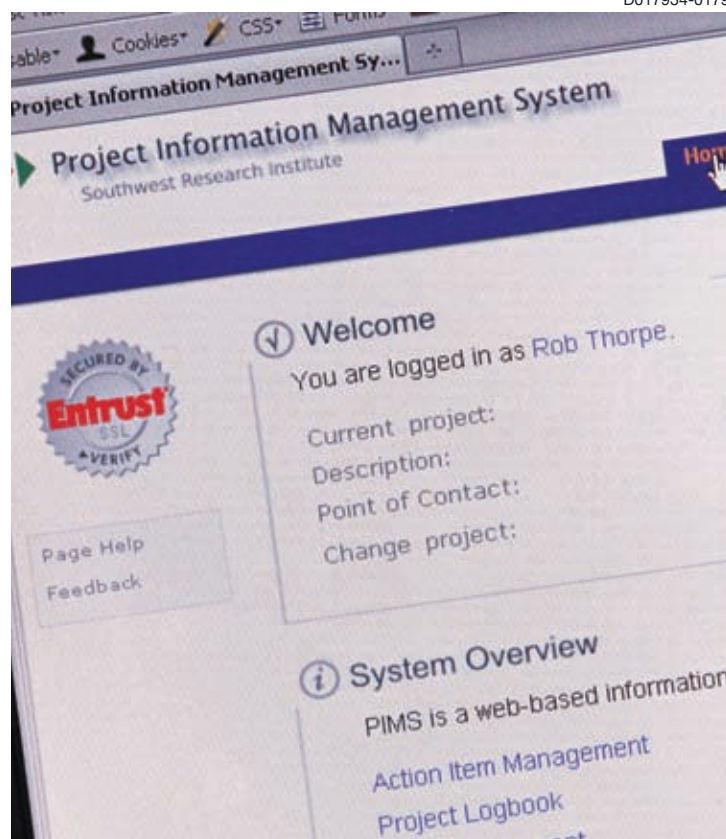
#### **TT: Was this a cost-effective approach?**

RT: Absolutely. It worked much better than spreadsheets and paper systems. Additionally, we designed custom features in each module to help streamline module-specific workflow issues.

For instance, in the Action Item module all action items can be categorized, sorted and queried in ways that can be customized for each project by the project manager. This is helpful when project managers want to filter action items by “Review” or “Institution,” or anything else they can think of. New categories often appear in the middle of the project, and the action item module is built to accommodate these changes.

The Risk Management module is designed to generate a PowerPoint presentation at the push of a few buttons. This greatly reduces the time-consuming, monthly task of reporting risk status to NASA headquarters.

Having an electronic Non-Conformance system might be the biggest time-saver of all. It uses email reminders to notify the next person in the approval chain when it's their turn to approve or disapprove a non-conformance,



*The Project Information Management System (PIMS) home page.*

and the email contains a link so users can jump right to the non-conformance web page. This formal approval process may involve up to 14 different electronic signatures, including those of clients.

Having email reminders and web-based non-conformance reports speeds this process up. One project manager said he easily saved \$50,000 a year using the electronic non-conformance system over the (now retired) paper-based one.

#### **TT: What is PIMS like today? How has it been accepted?**

RT: PIMS came into being in 2004. Now, seven years later, it is composed of 10 unique project management modules, all designed by project managers for project managers. It can support thousands of different projects simultaneously, and is used by NASA, ESA and commercial clients around the world.

#### **TT: Were there any surprises along the PIMS evolutionary path?**

RT: Several things happened that we didn't expect. One was that PIMS began to be referenced in proposals to promote project management capabilities. It helped that PIMS was well regarded by NASA. Gibson, who regularly serves on NASA review panels, relayed that it's continuously graded as a strength by proposal evaluation teams.

Additionally, customers began requesting their own PIMS servers. We worked with SwRI's legal department and created a “licensing” approach for PIMS. It now resides at two corporate sites off-campus, where we maintain and update the software and the customer takes care of the hardware and backups.

Today, PIMS is offered free to SwRI project teams and clients. It's sold on an annual “subscription” basis to external clients, who use the main SwRI PIMS site, and via a “licensed” service agreement

for clients who want their own on-site PIMS system.

A third side effect of PIMS was that more requests came in for new web applications. That led to the development of the Template System.

**TT: What is the Template System?**

RT: As we developed PIMS and other web applications we also made architecture improvements along the way. For instance, we found ways to make navigation easier, and enhanced usability. To preserve these ideas for future projects, we developed a template web application.

This fully functioning system has all the basic functions — login, user management, security, etc. When a new project comes along, the template system is designed to be cloned in a few days. After that, it's ready for software engineers to plug in project-specific functions. This is much more efficient than building a new web system, or taking an existing one and altering it, which can take weeks or even months.

For example, the stock management software program was 20 years old and struggling to manage hundreds of thousands of records, which it was never designed to do. The template system was cloned in a few days, and the complete set of stock management functions, from ordering to issuing parts, was implemented as the Project Support System (PSS). The implementation took quite some time — there were a lot of features that needed to be duplicated and a lot of testing to be done — but we



*The Template System provides for rapid development of custom web applications, such as this airline-style manufacturing status page installed in SwRI's High-voltage Power Supply Laboratory.*

knew we had a solid architecture from the template system and were able to focus our efforts on implementation.

**TT: How do you make sure these systems work OK?**

RT: That's really the trick. It's very important to have a strong test capability when developing reliable production-quality systems such as PIMS.

PIMS grew as new modules were added and users thought of new ways to make their jobs more efficient. At the same time, the PIMS test plan expanded into a large document, hundreds

of pages in length. The testing process was getting bigger and taking longer.

Again internet technology evolved to make testing more efficient. Java tools became available to allow development of a web application that could test other web applications. The Template System was cloned and functionality was added to create PIXEL, the Production Integrated XML Evaluator, which can test other web applications.

For example, after a software engineer develops a new function in PIMS, a test engineer creates a test script in PIXEL to check the function. Once that script is written, it can be used over and over again for every subsequent release of PIMS. Hundreds of these tests and tens of thousands of script commands can be executed at the push of a button.

The paper test plan still exists in a smaller form, because it's always good to have a real person looking at the system and running some tests. With PIXEL checking the details and users running a short test plan we have a reliable and easily repeatable test approach — that's very helpful.

**TT: What's next for PIMS, PSS and PIXEL?**

RT: The Internet continues to evolve and grow. As a web-application development team, our job is to evaluate technologies as they mature and offer practical advantages to customers and integrate them into our systems. PIMS, PSS and our other systems continue to grow as users and managers think of new ways to improve the quality of their work. The Template System and PIXEL help us keep the costs down and focus our efforts on delivering the capabilities the client wants.

It's quite a payoff for a friendly wager made back in 1998.

*Questions about this article? Contact Thorpe at (210) 522-2848 or robert.thorpe@swri.org.*

LSW - PIMS 5x5 Risk Matrix

		Consequence				
		1 Minimal	2 Moderate	3 Significant	4 Major	5 Unacceptable
5 Very High	5	Green	Yellow	Red	Red	Red
	4 High	Green	Yellow	Yellow	Red	Red
	3 Medium	Green	Green	Yellow	Yellow	Yellow
	2 Low	Green	Green	Green	Yellow	Yellow
	1 Very Low	Green	Green	Green	Green	Yellow

		Technical or Science Impact		
Consequence	Cost Impact	Schedule Impact	Technical or Science Impact	
5 Unacceptable	Exceeds project reserves	Affects primary and backup launch dates	Total loss of spacecraft or instruments, or failure to achieve all Level 1 science requirements	
4 Major	Exceeds segment reserves	Affects primary launch date, but not backup launch date	Major loss of spacecraft or instrument capability, or failure to achieve some Level 1 science requirements	
			Significant loss of spacecraft or instrument capability, or failure to achieve some Level 2 science objectives	

*The PIMS risk module provides risk management functions built around the NASA process, including the NASA 5x5 risk matrix.*

D017934-0177